



Mock exam
Programmierung

Summer Term 2019

Montag, den 15.07.2019

Exercise 1. (10 Points)



- (a) Allocate dynamically memory for 20 variables of type `double`.

- (b) Explain the terms lifetime (Lebensdauer) and visibility (Sichtbarkeit) of variables. Briefly talk about global and `static` variables.

- (c) Define a structure (with typedef) `student`, which includes a field `name` of type `char[20]` and a field `matrikelnummer` of type `int`. Deklare a variable of type `student` und assign your name and Matrikelnummer to it.

- (d) Explain the term buffer overflow.

- (e) Which advantages arise from using a linked list (verkettete Liste) instead of an array for storing data?

Exercise 2. (5 Points)

Explain how command line arguments in C can be interpreted utilising `getopt`.

Exercise 3. (6 Points)

- (a) Write a function `random_double`, which takes a parameter `double dmax` and a random number of type `double` in the range from 0 to `dmax`.

- (b) Write a procedure which takes a variable `int n` and recursively computes the factorial `n!` .

Exercise 4. (6 Points)

- (a) Write a procedure, which obtains a string as a parameter and replaces capital case with lower case letters and vice versa.

- (b) Write a procedure, which obtains two strings as parameters as well as a `char` array and its length. Append the content of the second string to the first and write the result into the `char` array. Output an error message in case that the array turns out to be too small.

- (c) Write a procedure, which takes a `char` array as a parameter, as well as two strings, which include a word each. Search for the first word in the array and replace it by the second. You may assume that the array is large enough.

Exercise 5. (6 Points)

(a) Define a data type `node`, which represents a node of a singly linked list (einfach verkettete Liste) and a data type `tree_node`, representing the node of a binary tree (binärer Baum). What is the main difference between the two?

(b) Assume a procedure

```
void print_tree_node(tree_node*);
```

has been implemented already, which outputs the content of the node of a binary tree. Write a procedure which takes a pointer to the root node of the tree and outputs the whole tree recursively. It does not matter in which order the nodes are printed, but all nodes shall be outputted.

Exercise 6. (6 Points)

(a) Define a new type `employee`. It shall be given by a structure which includes name, prename and salary. Choose a `char` array for the first two fields and an `int` for the salary.

(b) Write a function

```
employee* new_employee(char* name, char* prename, int salary);
```

It shall allocate dynamically memory for a new employee und set the name, prename and salary according to the parameters.

(c) Write a procedure

```
void print_employee(employee*, char* filename);
```

which opens a stream to a file called `filename`. This procedure shall append name, prename and salary to the file. Close the stream after the operation.